

Arquitetura Hexagonal

1. Introdução

Sabemos que na arquitetura hexagonal, temos a importante separação do sistema, em classes de domínio, adaptadores, portas e tecnologias externas. O intuito de haver todos esses elementos, é justamente separar o domínio do restante do sistema. O intuito de separar o domínio do restante do sistema é justamente que os desenvolvedores foquem no domínio, pois o domínio é a parte principal do sistema. A vista disso, um segundo benefício de adotar essa arquitetura é justamente a testabilidade além de que, com um domínio livre de tecnologia, é mais fácil também trocar de tecnologia posteriormente.

2. Aplicação da Arquitetura hexagonal no AdminFood

O Adminfood é um sistema criado para gerenciamento de restaurantes, assim tendo como definir muito bem o domínio, a partir do backlog do produto. Dito isso, o enfoque dos desenvolvedores foi de justamente garantir que as funcionalidades do domínio seriam atendidas, sendo elas:

- Criar o esquema de banco de dados
- Criar tabelas das principais entidades
- Criar a tela de login
- Criar a validação para os dados inseridos no login
- Criar a tela de cadastro de itens
- Criar método de cadastro dos itens de pedido
- Criar método de cadastro dos itens de estoque

Dito isso, como foi adotado uma metodologia ágil, os desenvolvedores foram entregando periodicamente os serviços do domínio, entretanto, houve um atraso no desenvolvimento que impossibilitou de todos os serviços serem entregues. A vista disso, em determinados serviços do sistema, como medida provisória, foi-se adotado a implementação de mini API's (adptadores) dentro do domínio, mesmo sendo contra os princípios da arquitetura hexagonal. Esses adaptadores, foram forçadamente inseridos como meio de agilizar a implementação, entretanto, numa próxima atualização do sistema (Sprint 3), tem-se como foco modularizar o sistema em sua plenitude, dentro da arquitetura hexagonal, sem adaptações forçadas. Um outro compromisso que os desenvolvedores assumirão para o próximo Sprint como melhoria do sistema, é criar um EndPoint para plotar os gráficos de desempenho do restaurante por meio de uma API (adaptador) e uma porta de entrada do sistema, seguindo os padrões da arquitetura hexagonal.